

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A method for maintaining timers for a computer system, the method comprising:

(a) writing a plurality of timer values in a connection table, wherein each timer value ~~indicating~~ indicates a timeout for a ~~particular~~ different timer, and each timer is associated with one of a plurality of connections of the computer system, and wherein each one of the connections is associated with a plurality of the timers ~~the connection table includes a plurality of timers for each connection;~~ and

(b) writing one of the timer values for each of the connections to a global timer array ~~for each connection~~, such that the global timer array can be scanned to determine when timeouts occur for active connections to the computer system.

2. (Currently amended) The method of claim 1 wherein each of the timer value values written to the global timer array for each connection is associated with the timer that will expire earliest out of the plurality of timers ~~for~~ associated with that ~~the~~ connection.

3. (Currently amended) The method of claim 2 wherein ~~the~~ each timer value is a timestamp indicating at what time the timeout will occur.

4. (Original) The method of claim 3 wherein the timestamp is compared to the current time

to determine if the timeout has occurred.

5. (Currently amended) The method of claim 1 wherein if a timer value changes, ~~the~~ a new timer value is written to the connection table, and if the new timer value would expire sooner than the associated timer value stored in the global timer array, the new timer value is written into the global timer array in place of the associated timer value.

6. (Original) The method of claim 1 further comprising periodically running a timer task to examine the global timer array for timeouts.

7. (Original) The method of claim 6 wherein when a timeout is found in the global timer array, the associated timer value in the connection table is referenced and the functions associated with the timeout are initiated.

8. (Original) The method of claim 1 wherein the timer value is a counter value that is periodically incremented or decremented until a predetermined value is reached to indicate a timeout.

9. (Original) The method of claim 1 wherein the plurality of timers for each connection include an idle timer for determining an idle time of a connection, a keepalive timer for determining if a connected computer system is responding, and a retransmission timer for determining when data should be retransmitted to the connected computer system.

10. (Original) The method of claim 1 wherein the timers of a connection are associated with a particular communication protocol used for that connection.

11. (Original) The method of claim 10 wherein the communication protocol is Transmission Control Protocol (TCP).

12. (Original) The method of claim 1 wherein at least one timer value is written to the connection table when a timeout or network activity occurs.

13. (Currently amended) A method for examining timers for a computer system, the method comprising:

(a) scanning a plurality of array timer values for timeouts, the array timer values stored in a timer array and each array timer value representing a timer, wherein a single array timer value is stored in the timer array for each one of a plurality of active network connection to the computer system, and wherein each one of the active network connections is associated with a plurality of timers; and

(b) causing a computer system function to be initiated when a timeout is indicated by a particular array timer value associated with a particular active connection, the computer system function being associated with one of ~~a plurality of~~ the timers used for the particular active connection.

14. (Currently amended) The method of claim 13 wherein the single array timer value for each active connection is associated with the timer that will expire earliest out of the timers for the

active connection.

15. (Original) The method of claim 13 wherein when the timeout is indicated, at least one of a plurality of table timer values is referenced in a connection table, each table timer value being associated with one of the timers used for the particular active connection.

16. (Original) The method of claim 15 wherein the referenced table timer value matches the array timer value indicating the timeout.

17. (Original) The method of claim 13 further comprising writing the earliest of the table timer values for the active connection to the timer array.

18. (Original) The method of claim 15 wherein the array timer values and the table timer values are timestamps indicating at what time each timeout will occur, and wherein a timestamp is compared to the current time to determine if a timeout has occurred for that timestamp.

19. (Original) The method of claim 15 wherein the array timer values and the table timer values are each a counter value that is periodically incremented or decremented until a predetermined value is reached to indicate a timeout.

20. (Original) The method of claim 15 wherein the plurality of table timer values for the active connection include an idle timer value for determining an idle time of the connection, a keepalive timer value for determining if a connected computer system is responding, and a

retransmission timer value for determining when data should be retransmitted to the connected computer system.

21. (Original) The method of claim 13 wherein the timers of a connection are associated with a particular communication protocol used for that connection.

22. (Original) The method of claim 13 wherein the scanning of array timer values includes scanning only a subset of the array timer values, and wherein a different subset of the array timer values is scanned in each one of a plurality of processes scanning the array in parallel.

23. (Original) A method for restarting a timer for a computer system, the method comprising:

(a) starting a timer, the timer for determining when a timeout occurs for an associated network connection of the computer system;

(b) restarting the timer if data is received from or transmitted to the connected computer system before the timeout occurs and after a predetermined time interval after the timer is started; and

(c) not restarting the timer if data is received from or transmitted to the connected computer system before the timeout occurs and within the predetermined time interval after the timer is started.

24. (Original) The method of claim 23 further comprising initiating a computer system function if the timeout occurs.

25. (Original) The method of claim 24 wherein the timer is an idle timer for determining whether a timeout occurs after a period of time in which no communication occurs on the network connection associated with the idle timer.

26. (Original) The method of claim 24 wherein the computer system function includes making the network connection inactive.

27. (Original) The method of claim 23 wherein the timer includes a timer value that is a timestamp indicating at what time the timeout will occur, and wherein the timestamp is compared to the current time to determine if the timeout has occurred.

28. (Original) The method of claim 23 wherein the timer includes a timer value that is a counter value that is periodically incremented or decremented until a predetermined value is reached to indicate a timeout.

29. (Original) The method of claim 23 wherein the timer includes a timer value that is stored in a connection table storing all the timer values for all network connections to the computer system.

30. (Original) The method of claim 23 wherein the timer includes a timer value that is stored in a global timer array, the timer value associated with the timer that will expire earliest out of a plurality of timers for the network connection.

31. (Original) The method of claim 23 wherein the timer is associated with a particular communication protocol used for that connection, wherein the communication protocol is Transmission Control Protocol (TCP).

32. (Currently amended) A computer readable medium including program instructions to be implemented by a computer, the program instructions for maintaining timers for a computer system, the program instructions implementing steps comprising:

(a) writing a plurality of timer values in a connection table, wherein each timer value ~~indicating~~ indicates a timeout for a ~~particular~~ different timer, and each timer is associated with one of a plurality of connections of the computer system, and wherein each one of the connections is associated with a plurality of the timers ~~the connection table includes a plurality of timers for each connection;~~ and

(b) writing one of the timer values for each of the connections to a global timer array for each connection, such that the global timer array can be scanned to determine when timeouts occur for active connections to the computer system.

33. (Currently amended) The computer readable medium of claim 32 wherein each of the timer ~~value~~ values written to the global timer array for each connection is associated with the timer that will expire earliest out of the plurality of timers for associated with that ~~the~~ connection.

34. (Currently amended) The computer readable medium of claim 33 wherein ~~the~~ each timer value is a timestamp indicating at what time the timeout will occur, wherein the timestamp is compared to the current time to determine if the timeout has occurred.

35. (Currently amended) The computer readable medium of claim 32 wherein if a timer value changes, ~~the~~ a new timer value is written to the connection table, and if the new timer value would expire sooner than the associated timer value stored in the global timer array, the new timer value is written into the global timer array in place of the associated timer value.

36. (Currently amended) A system for maintaining timers for a computer system, the system comprising:

a connection table writer for writing a plurality of timer values in a connection table, wherein each timer value ~~indicating~~ indicates a timeout for a ~~particular~~ different timer, and each timer is associated with one of a plurality of connections of the computer system, and wherein each one of the connections is associated with a plurality of the timers ~~the connection table includes a plurality of timers for each connection;~~ and

a global array writer for writing one of the timer values for each of the connections to a global timer array ~~for each connection~~, such that the global timer array can be scanned to determine when timeouts occur for active connections to the computer system.

37. (Currently amended) The system of claim 36 wherein each of the timer value values written to the global timer array for each connection is associated with the timer that will expire earliest out of the plurality of timers ~~for~~ associated with that the connection.

38. (Currently amended) A computer readable medium including program instructions to be implemented by a computer, the program instructions for examining timers for a computer system,

the program instructions implementing steps comprising:

(a) scanning a plurality of array timer values for timeouts, the array timer values stored in a timer array and each array timer value representing a timer, wherein a single array timer value is stored in the timer array for each one of a plurality of active network connection to the computer system, and wherein each one of the active network connections is associated with a plurality of timers; and

(b) causing a computer system function to be initiated when a timeout is indicated by a particular array timer value associated with a particular active connection, the computer system function being associated with one of ~~a plurality of~~ the timers used for the particular active connection.

39. (Currently amended) The computer readable medium of claim 38 wherein the single array timer value for each active connection is associated with the timer that will expire earliest out of the timers for the active connection.

40. (Original) The computer readable medium of claim 38 wherein when the timeout is indicated, at least one of a plurality of table timer values is referenced in a connection table, each table timer value being associated with one of the timers used for the particular active connection.

41. (Currently amended) A system for examining and processing timers for a computer system, the system comprising:

a timer task for scanning a plurality of array timer values for timeouts, the array timer values stored in a timer array and each array timer value representing a timer, wherein a single array timer

value is stored in the timer array for each one of a plurality of active network connection to the computer system, and wherein each one of the active network connections is associated with a plurality of timers; and

a processing task for initiating a computer system function for a particular active connection when a timeout has been indicated by the timer task, the computer system function being associated with one of ~~a plurality of~~ the timers used for the particular active connection.

42. (Currently amended) The system of claim 41 wherein the single array timer value for each active connection is associated with the timer that will expire earliest out of the timers for the active connection.

43. (Original) The system of claim 41 wherein when the timeout is indicated, the processing task references at least one of a plurality of table timer values in a connection table, each table timer value being associated with one of the timers used for the particular active connection.

44. (Original) A computer readable medium including program instructions to be implemented by a computer, the program instructions for restarting a timer for a computer system, the program instructions implementing steps comprising:

(a) starting a timer, the timer for determining when a timeout occurs for an associated network connection of the computer system;

(b) restarting the timer if data is received from or transmitted to the connected computer system before the timeout occurs and after a predetermined time interval after the timer is started;
and

(c) not restarting the timer if data is received from or transmitted to the connected computer system before the timeout occurs and within the predetermined time interval after the timer is started.

45. (Original) A system for restarting a timer for a computer system, the system comprising:
a timer starter for starting a timer, the timer for determining when a timeout occurs for an associated network connection of the computer system; and
a timer restarter for restarting the timer if data is received from or transmitted to the connected computer system before the timeout occurs and after a predetermined time interval after the timer is started, wherein the timer is not restarted if data is received from or transmitted to the connected computer system before the timeout occurs and within the predetermined time interval after the timer is started.